

სასწავლო კურსის სახელწოდება:	მონაცემთა ბაზების ადმინისტრირება
ლექტორი:	ალექსანდრე ჩახვაძე
სტუდენტის სახელი და გვარი	
ჯგუფის #	

შუალედური გამოცდა

ნიმუში

გამოცდაზე მუშაობთ თქვენთვის განკუთვნილ ამ ფაილში, სათანადო ადგილას უთითებთ საჭირო ინფორმაციებს, სვამთ სკრიპტებსა და სქრინებს. გამოცდაზე მუშაობთ ვირტუალურ გარემოში არსებულ ბაზაზე. გამოცდაზე შეგიძლიათ გამოიყენოთ ფაილი: oca_dba_exam.txt . დასრულებულს ფაილს დაარქვით სახელი და გვარი, დააპედეფეთ და ატვირთეთ საგამოცდო პროგრამის სათანადო ველში.

შეფასება: 15 საკითხი ჯამში 25 ქულა, კერძოდ საკითხები 1,2,4,6,7,8,9,10,11,12 თითო 2 ქულა, ხოლო საკითხები 3,5,13,14,15 თითო 1 ქულა.

1. (2 ქულა) დაუკავშირდით ორაკლ სერვერს როგორც პრივილეგირებული მომხმარებელი system და შექმენით როლი სკრიპტულად თქვენი გვარი_role და მიანიჭეთ შემდეგი სისტემური პრივილეგიები create session, create any table, create any view, create any index, create any sequence, create any synonym, alter any table, alter any index, alter any sequence, drop any table, drop any view, drop any index, drop any sequence, drop any synonym. მიანიჭეთ ობიექტ პრივილეგია მოსელექტება hr სქემის employees ცხრილზე. მიანიჭეთ როლი dba შექმნილ როლს.

სკრიპტი:

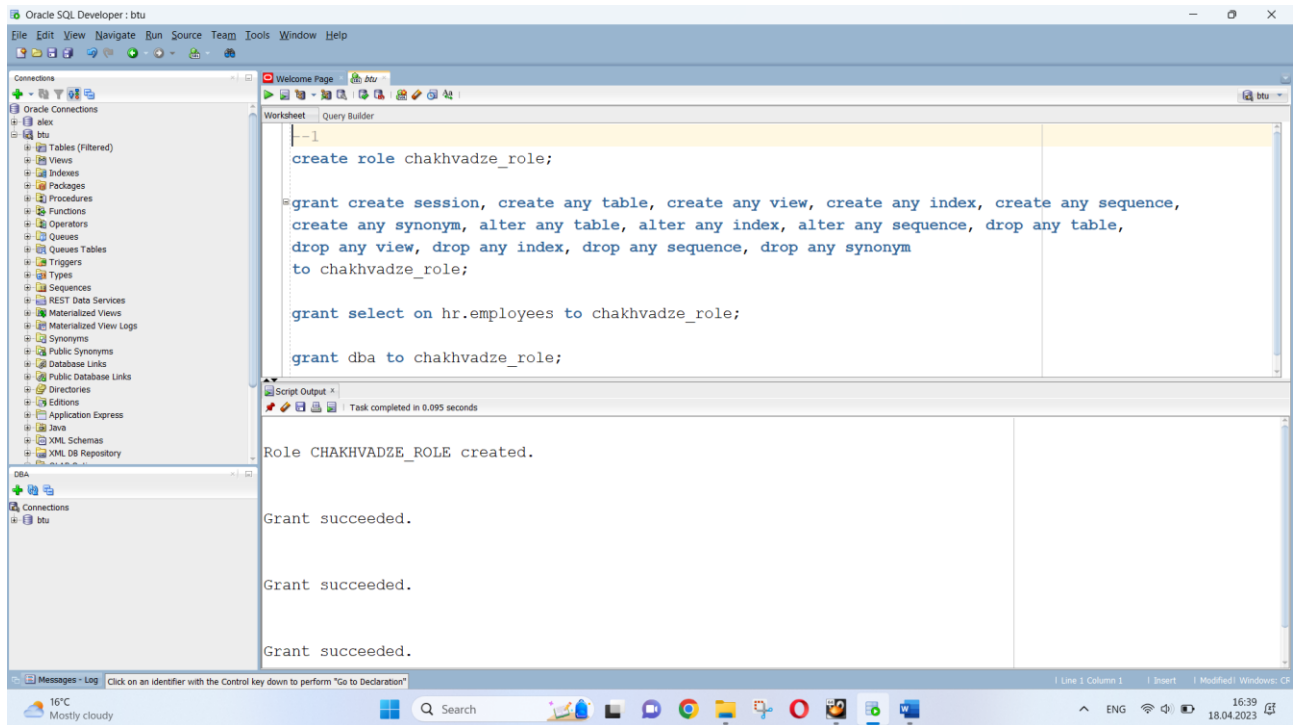
```
create role chakhvadze_role;
```

```
grant create session, create any table, create any view, create any index, create any sequence,
create any synonym, alter any table, alter any index, alter any sequence, drop any table,
drop any view, drop any index, drop any sequence, drop any synonym
to chakhvadze_role;
```

```
grant select on hr.employees to chakhvadze_role;
```

```
grant dba to chakhvadze_role;
```

სქრინი:



2. (2 ქულა) სკრიპტულად შექმენი მომხმარებელი **btu_გვარი** პაროლი **btu**, დეფაულტ ცხრილსივრცე **users** ულიმიტოდ, დროებით ცხრილსივრცედ **temp**, პროფაილი **default**. გადაამოწმეთ იუზერის არსებობა. მიანიჭეთ შექმნილი როლი ამ მომხმარებელს.

სკრიპტი:

```

create user btu_chakhvadze identified by btu
default TABLESPACE users QUOTA UNLIMITED on users
TEMPORARY TABLESPACE temp
profile default;
  
```

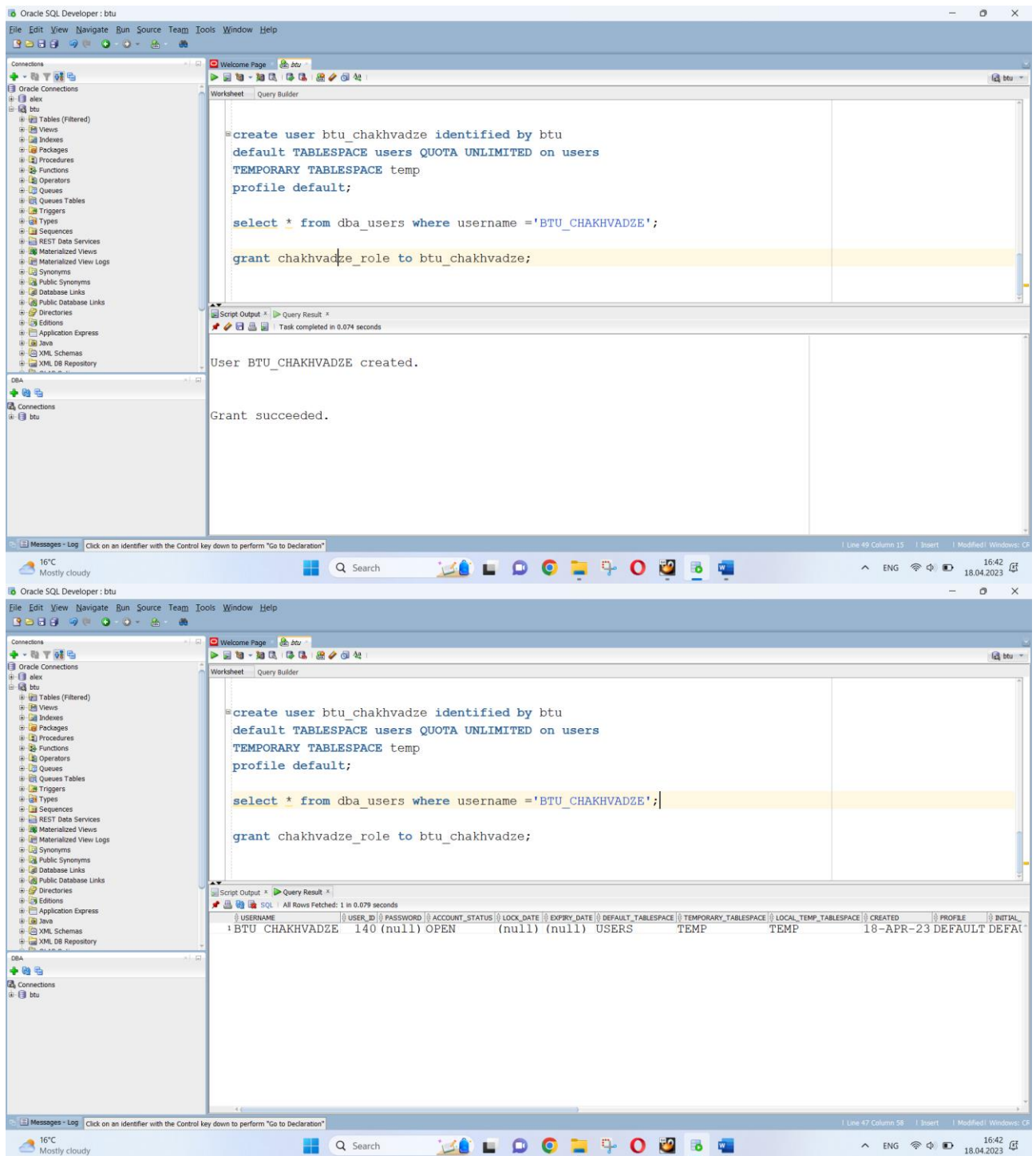
```

select * from dba_users where username = 'BTU_CHAKHVADZE';
  
```

```

grant chakhvadze_role to btu_chakhvadze;
  
```

სკრიპტი:



3. (1 ქულა) მიუთვლით ბაზას როგორც მომხმარებელი **btu_გვარი**. გადამოწმეთ ვინაობა. გადამოწმეთ თუ ფლობს რაიმე ობიექტს.

სკრიპტი:

show user

select * from dba_objects where owner='BTU_CHAKHVADZ';

სკრინი:

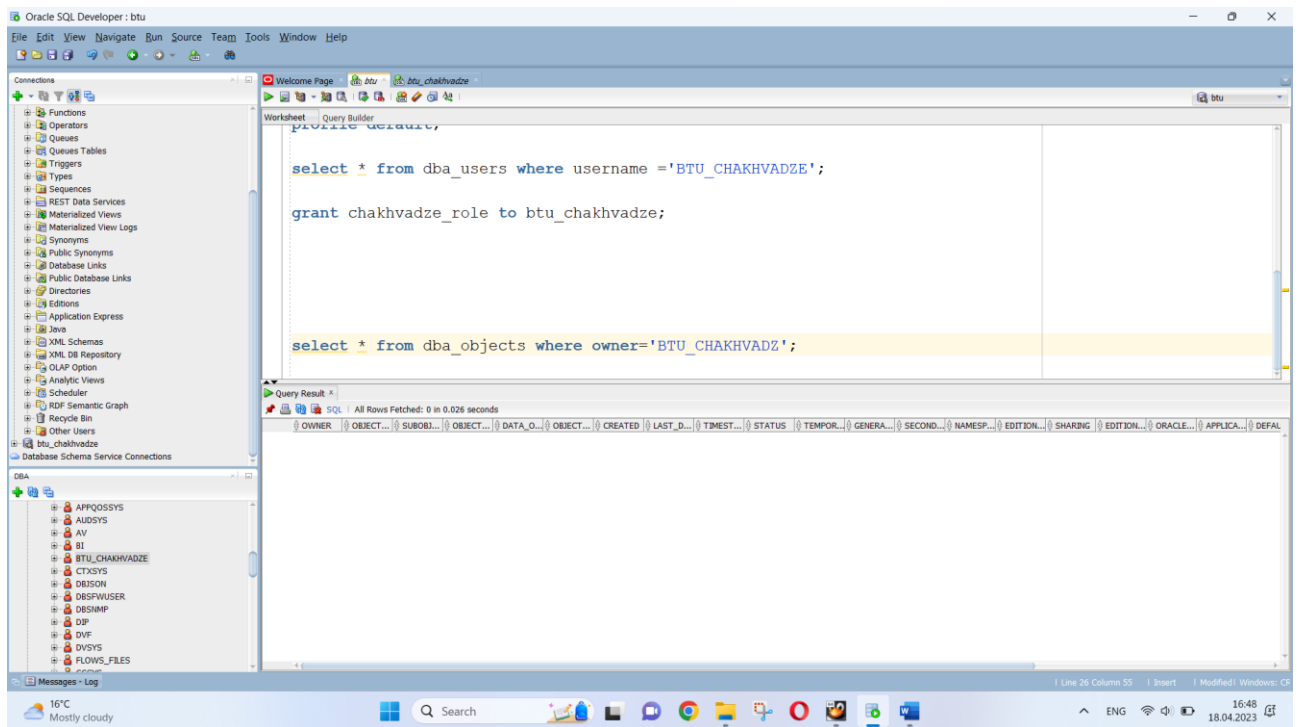
The image consists of two screenshots of the Oracle SQL Developer interface, demonstrating the process of connecting to a database and verifying a user.

Top Screenshot: New / Select Database Connection Dialog

- Connection Name:** btu_chakhvadze
- Database Type:** Oracle
- User Info:**
 - Authentication Type: Default
 - Username: btu_chakhvadze
 - Password: (masked with asterisks)
 - Role: default
 - ☒ Save Password
- Connection Type:** Basic
- Details:**
 - Hostname: localhost
 - Port: 1521
 - ☐ SID
 - ☒ Service name: orcl

Bottom Screenshot: SQL Worksheet and Script Output

- Worksheet:** Contains the SQL command `show user`.
- Script Output:** Shows the result of the command: `USER is "BTU_CHAKHVADZE"`.
- Database Schema Service Connections:** The left pane shows a tree of database schemas, including `BTU_CHAKHVADZE`.



4. (2 ქულა) შექმენი ცხრილი **test_გვარი** რომელსაც ექნება 4 სვეტი: პირველი id რიცხვითი, salary რიცხვითი (5,2) და მე-3 ვირტუალური სვეტი status, რომელიც შეყვანილი ხელფასის მიხედვით დააგენერირებს სვეტის მნიშვნელობას თუ შეტანილი იქნება salary-ში 500 ან მასზე ნაკლები low, თუ 500-ზე მეტი good, მე-4 სვეტი comm სიმბოლური მაქსიმუმ 30 სიმბოლო უჩინარი. შექმენი sal_idx ინდექსი salary სვეტზე.

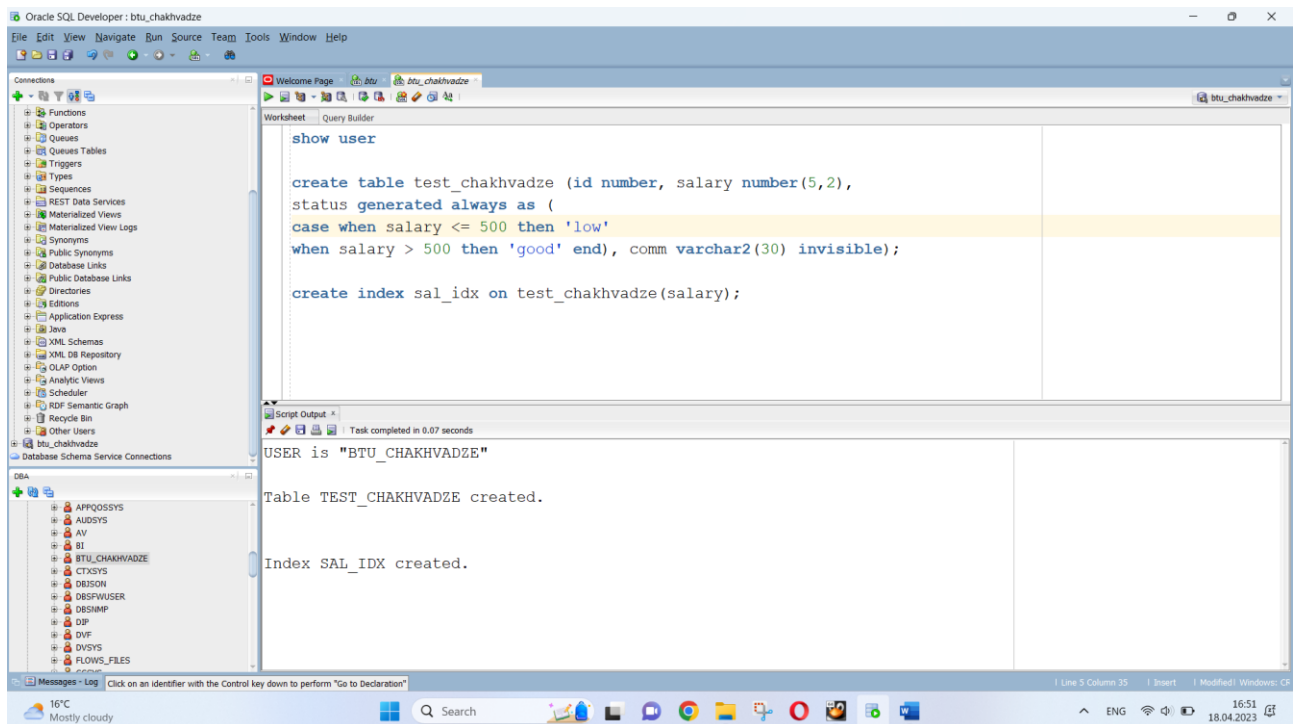
სკრიპტი:

```

create table test_chakhvadze (id number, salary number(5,2),
status generated always as (
case when salary <= 500 then 'low'
when salary > 500 then 'good' end), comm varchar2(30) invisible);
create index sal_idx on test_chakhvadze(salary);

```

სკრინი:



The screenshot shows the Oracle SQL Developer interface. The left pane displays the database schema with the user 'BTU_CHAKHVADZE' selected. The main window shows the following SQL script:

```
show user

create table test_chakhvadze (id number, salary number(5,2),
status generated always as (
case when salary <= 500 then 'low'
when salary > 500 then 'good' end), comm varchar2(30) invisible);

create index sal_idx on test_chakhvadze(salary);
```

The Script Output pane shows the following messages:

```
USER is "BTU_CHAKHVADZE"

Table TEST_CHAKHVADZE created.

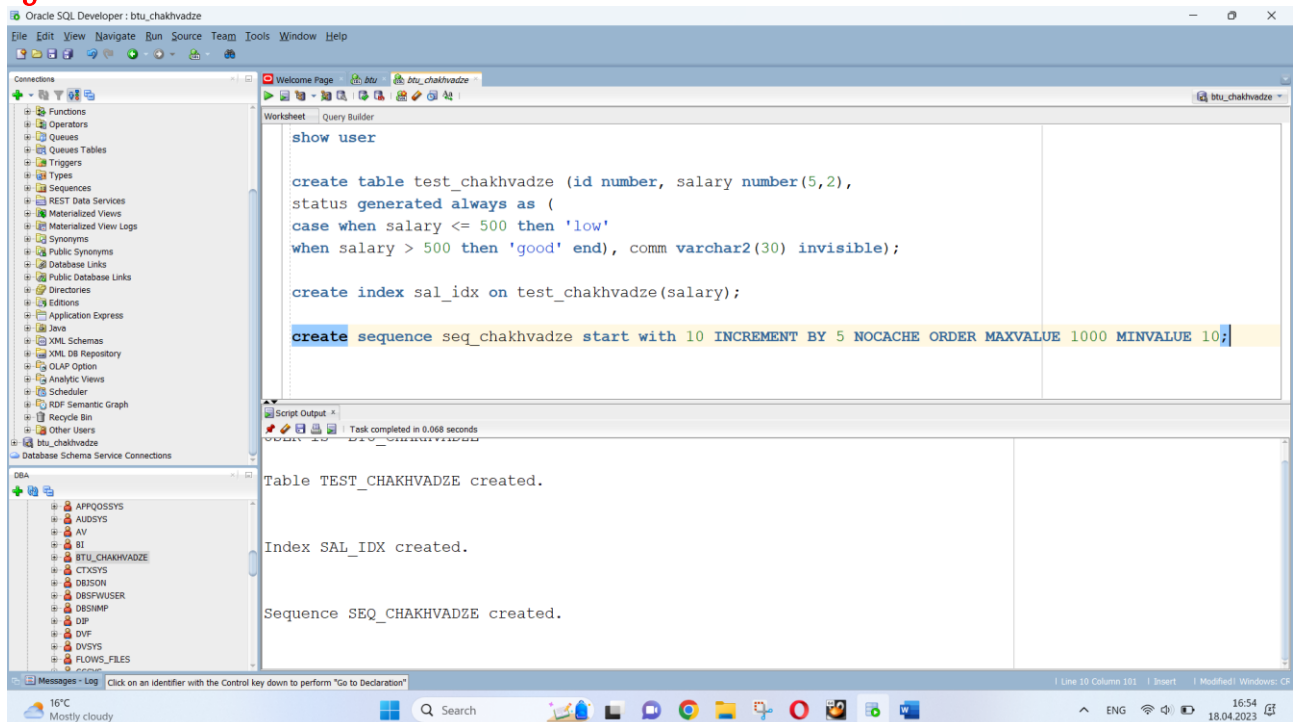
Index SAL_IDX created.
```

5. (1 ქულა) შექმენით მიმდევრობა **seq_გვარი** სახელით რომლის სტარტის წერტილია 10, ბიჯი 5, არ არის ციკლური, არის მოწესრიგებული, არ იყოს ქეშირებული, მაქსიმალური მნიშვნელობა 1000 და მინიმალური მნიშვნელობა 10.

სკრიპტი:

create sequence seq_chakhvadze start with 10 INCREMENT BY 5 NOCACHE ORDER MAXVALUE 1000 MINVALUE 10;

სკრინი:



The screenshot shows the Oracle SQL Developer interface. The left pane displays the database schema with the user 'BTU_CHAKHVADZE' selected. The main window shows the following SQL script:

```
show user

create table test_chakhvadze (id number, salary number(5,2),
status generated always as (
case when salary <= 500 then 'low'
when salary > 500 then 'good' end), comm varchar2(30) invisible);

create index sal_idx on test_chakhvadze(salary);

create sequence seq_chakhvadze start with 10 INCREMENT BY 5 NOCACHE ORDER MAXVALUE 1000 MINVALUE 10;
```

The Script Output pane shows the following messages:

```
USER is "BTU_CHAKHVADZE"

Table TEST_CHAKHVADZE created.

Index SAL_IDX created.

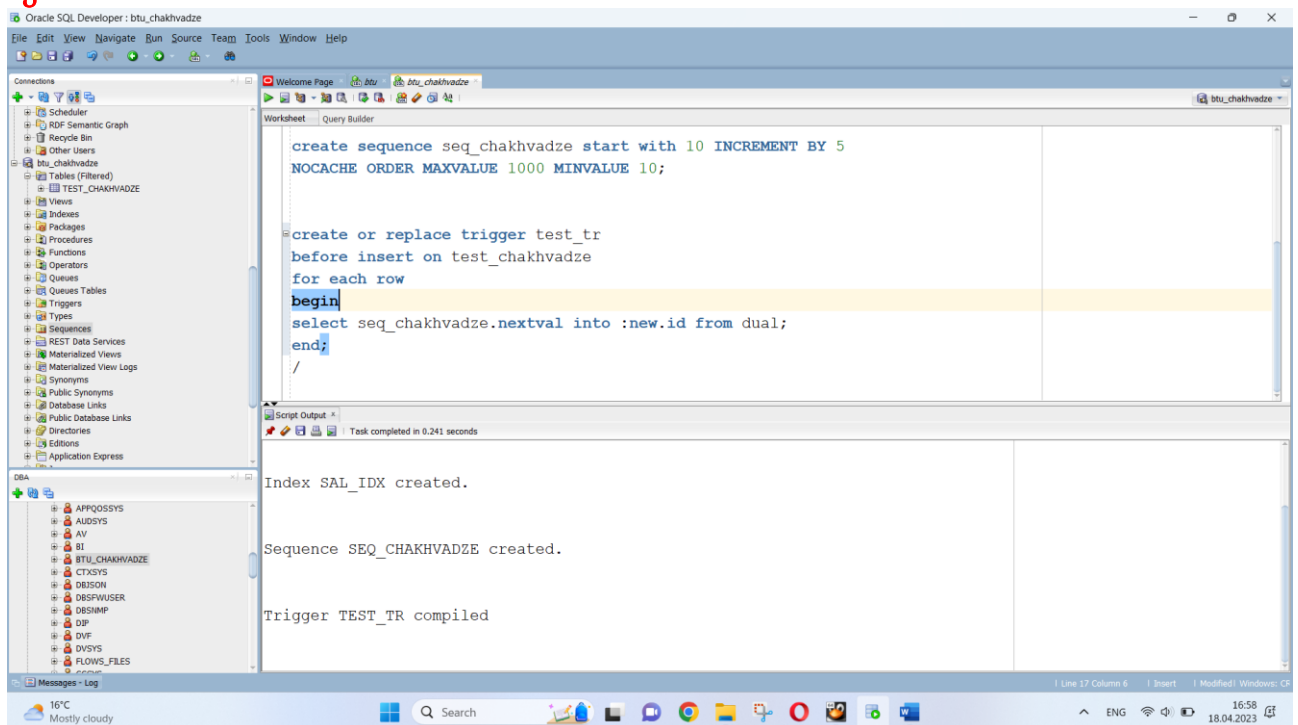
Sequence SEQ_CHAKHVADZE created.
```

6. (2 ქულა) შექმენი ტრიგერი test_tr რომელიც ავტომატურ ზრდად id სვეტის მნიშვნელობებს დააგენერირებს ახალი ჩანაწერის შეტანისას test_გვარი ცხრილში.

სკრიპტი:

```
create or replace trigger test_tr
before insert on test_chakhvadze
for each row
begin
select seq_chakhvadze.nextval into :new.id from dual;
end;
/
```

სკრინი:



7. (2 ქულა) შეიტანეთ test_chakhvadze ცხრილში შემდეგი ჩანაწერები salary და comm სვეტებში

450, 'btu is the best'

120, 'cool exam'

150, 'this day'

580, 'perfect thing'

890, 'good idea'

900, 'great option'

სკრიპტი:

```
insert into test_chakhvadze(salary, comm) values (450, 'btu is the best');
```

```
insert into test_chakhvadze(salary, comm) values (120, 'cool exam');
```

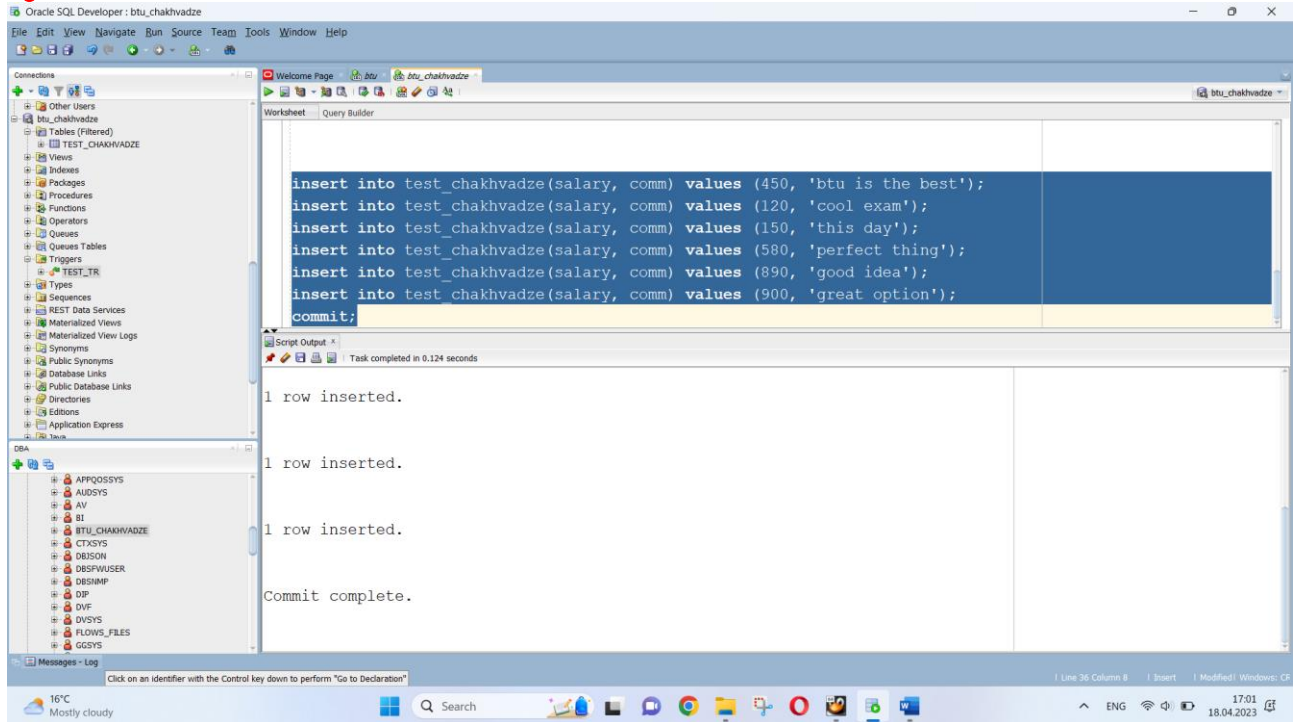
```
insert into test_chakhvadze(salary, comm) values (150, 'this day');
```

```
insert into test_chakhvadze(salary, comm) values (580, 'perfect thing');
```



```
insert into test_chakhvadze(salary, comm) values (890, 'good idea');
insert into test_chakhvadze(salary, comm) values (900, 'great option');
commit;
```

სკრინი:



8. (2 ქულა) შექმენით ქვემოთხოვნიტ ცხრილი emp სქემა hr-ის employees-ზე დაყრდნობით, საიდანაც აიღება მხოლოდ ის ჩანაწერები სადაც გვარის სიგრძე არის 7-სიმბოლოზე მეტი. შექმენით ქვემოთხოვნიტ ცხრილი emp2 ცხრილ emp-ზე დაყრდნობით მხოლოდ კონსტრუქცია და არა ჩანაწერებიც.

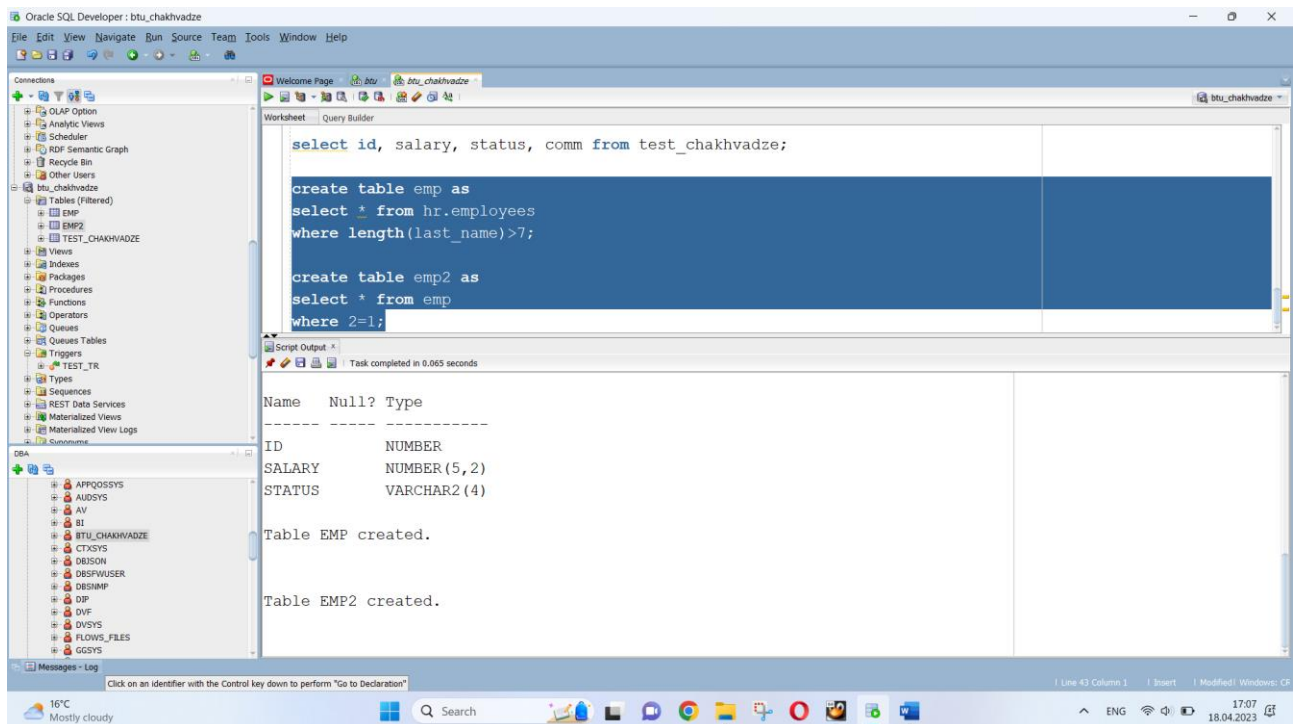
სკრიპტი:

```
create table emp as
select * from hr.employees
where length(last_name)>7;
```

```
create table emp2 as
select * from emp
where 2=1;
```

```
select * from emp;
select * from emp2;
```

სკრინი:



The screenshot shows the Oracle SQL Developer interface. The left pane displays the database schema with tables EMP and EMP2. The main window shows the following SQL queries:

```
select id, salary, status, comm from test_chakhvadze;
```

```
create table emp as
select * from hr.employees
where length(last_name)>7;
```

```
create table emp2 as
select * from emp
where 2=1;
```

The Script Output pane shows the following messages:

```
Task completed in 0.065 seconds
```

```
Name Null? Type
-----
ID      NUMBER
SALARY  NUMBER(5,2)
STATUS  VARCHAR2(4)
```

```
Table EMP created.
```

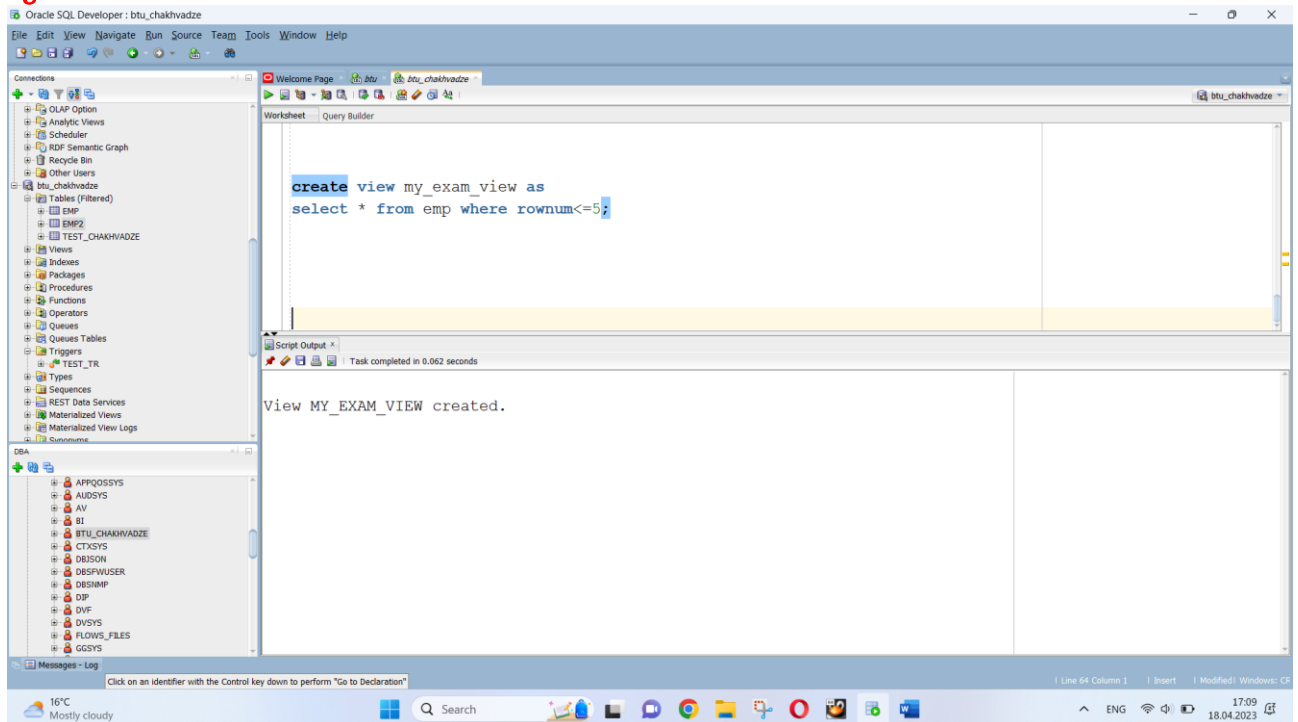
```
Table EMP2 created.
```

9. (2 ქულა) შექმენი **my_exam_view** ხედი emp ცხრილის პირველი 5 ჩანაწერით.

სკრიპტი:

create view my_exam_view as
select * from emp where rownum<=5;

სკრინი:



The screenshot shows the Oracle SQL Developer interface. The left pane displays the database schema with tables EMP and EMP2. The main window shows the following SQL query:

```
create view my_exam_view as
select * from emp where rownum<=5;
```

The Script Output pane shows the following message:

```
Task completed in 0.062 seconds
```

```
View MY_EXAM_VIEW created.
```

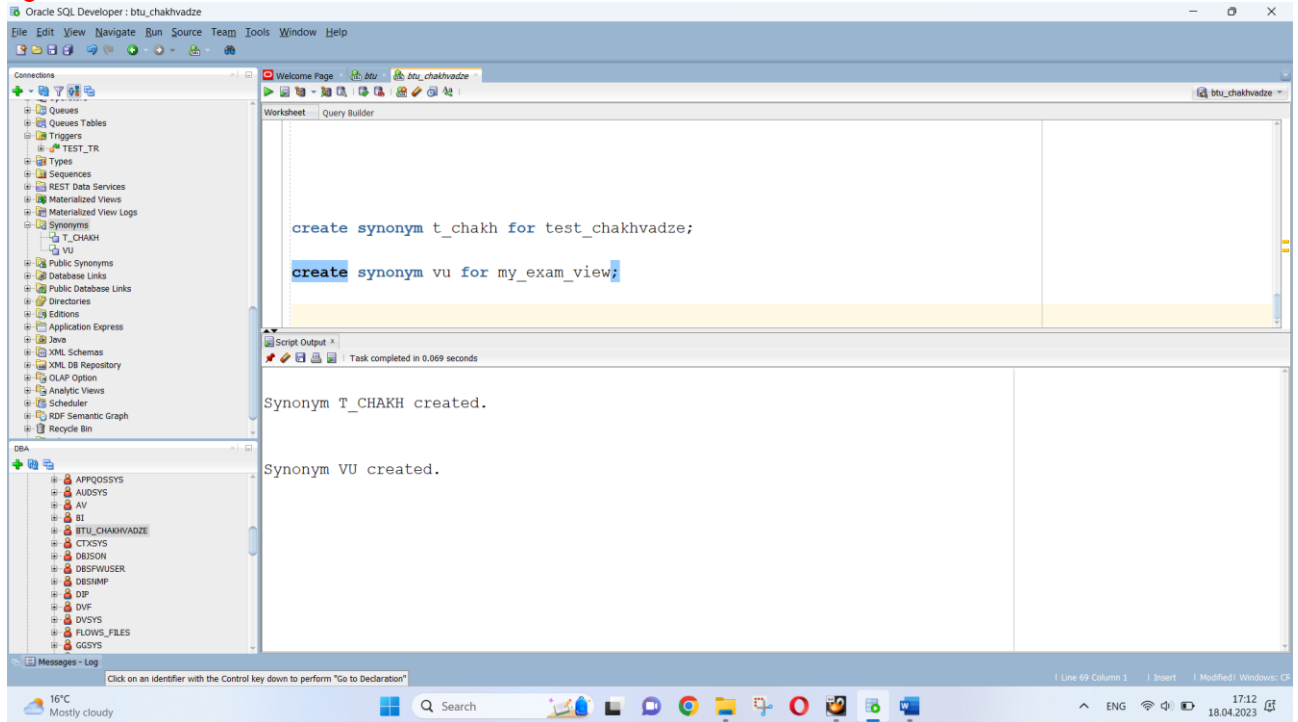
10. (2 ქულა) შექმენი სინონიმები **test_chakhvadze** ცხრილზე t_chakh (t_გვარის პირველი 3 ასო) სახელით და **my_exam_view** ხედზე vu სახელით. გადაამაწმეთ თუ რა ობიექტებს ფლობს მომხმარებელი.

სკრიპტი:

create synonym t_chakh for test_chakhvadze;

create synonym vu for my_exam_view;

სკრინი:



11. (2 ქულა) a) შეიტანეთ ახალი ჩანაწერი ცხრილში test_chakhvadze 300, 'gtu forever'; b) დაამოდიფიცირეთ ჩანაწერი რომლის აიდეა 20 comm სვეტში არსებულს მიუკონკატენირეთ თრანქირებული სისტემური თარიღი; c) წაშალოთ ის სტრიქონები რომლებშიც comm სვეტის სიგრძე არის 2-ის ჯერადი; d) დაამატეთ სვეტი manager რიცხვითი; e) სვეტი manager გახადეთ სიმბოლური მაქსიმალური 50 სიმბოლო; f) გადარქვით სახელი manager სვეტს man სახელით; g) ამოშალოთ სვეტი man; h) გახადეთ გამოუყენებელი comm; i) გადარქვით ცხრილს test_chakhvadze სახელი დაარქვით chakhva_tab; j) გახადეთ ცხრილი მხოლოდ ამოკითხვადი.

სკრიპტი:

a)

insert into test_chakhvadze(salary, comm) values (300, 'gtu forever');
commit;

b)

update test_chakhvadze set comm=comm||trunc(sysdate)
where id = 20;
commit;

c)

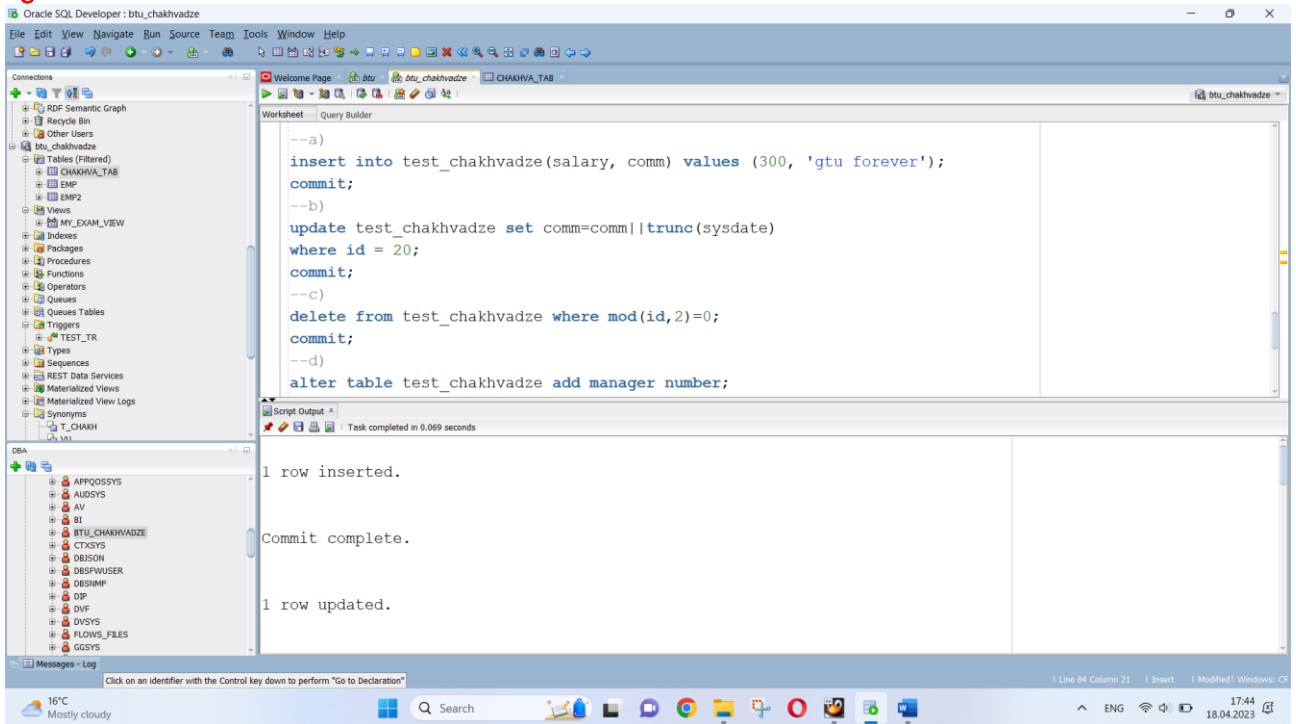
delete from test_chakhvadze where mod(id,2)=0;
commit;

d)

alter table test_chakhvadze add manager number;

- e)
alter table test_chakhvadze MODIFY manager varchar2(50);
- f)
alter table test_chakhvadze rename column manager to man;
- g)
alter table test_chakhvadze drop column man;
- h)
alter table test_chakhvadze set UNUSED(comm);
- i)
alter table test_chakhvadze RENAME to chakhva_tab;
- j)
alter table chakhva_tab read only;

სკრიპტი:

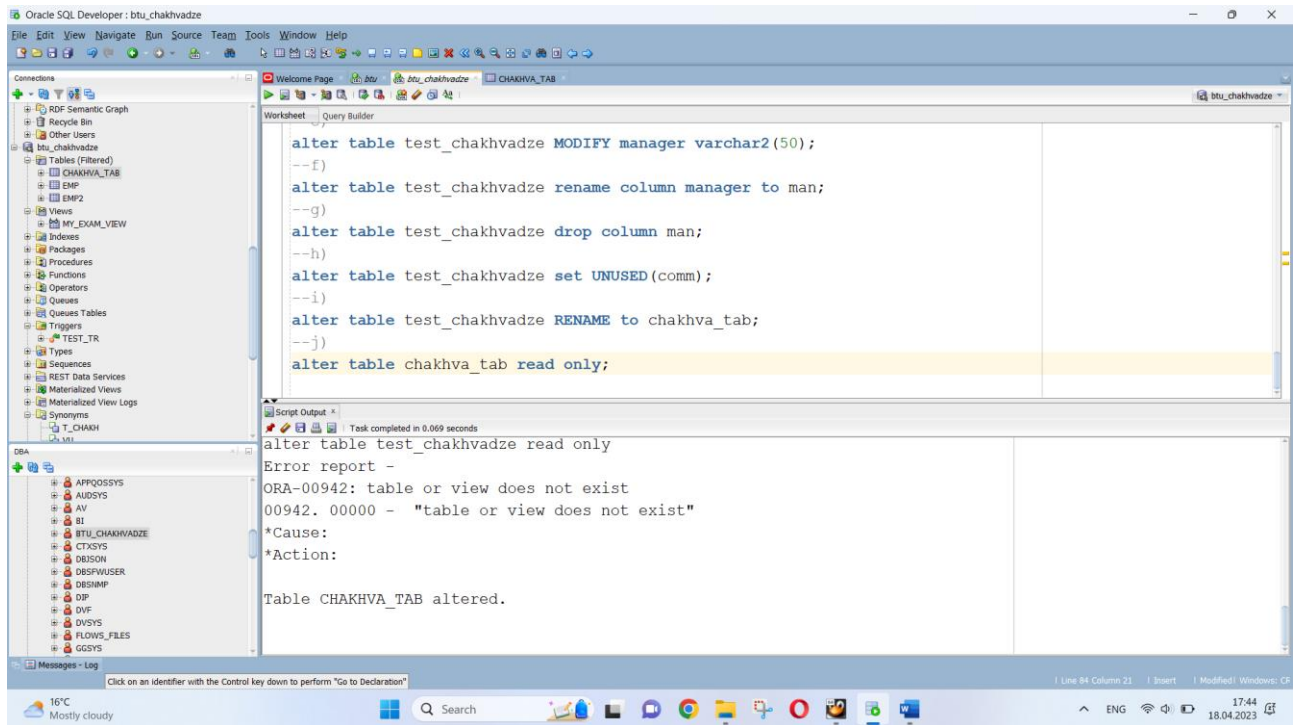


```
--a)
insert into test_chakhvadze(salary, comm) values (300, 'gtu forever');
commit;
--b)
update test_chakhvadze set comm=comm||trunc(sysdate)
where id = 20;
commit;
--c)
delete from test_chakhvadze where mod(id,2)=0;
commit;
--d)
alter table test_chakhvadze add manager number;
```

1 row inserted.

Commit complete.

1 row updated.



12. (2 ქულა) შექმენით ახალი პროფილი **exam_prof_გვარი** რომელსაც ექნება შემდეგი პარამეტრები

COMPOSITE_LIMIT	default
SESSIONS_PER_USER	30
CPU_PER_SESSION	unlimited
CPU_PER_CALL	unlimited
LOGICAL_READS_PER_SESSION	400000
LOGICAL_READS_PER_CALL	10000
IDLE_TIME	8
CONNECT_TIME	200
PRIVATE_SGA	default
FAILED_LOGIN_ATTEMPTS	20
PASSWORD_LIFE_TIME	100
PASSWORD_REUSE_TIME	140
PASSWORD_REUSE_MAX	15
PASSWORD_VERIFY_FUNCTION	null
PASSWORD_LOCK_TIME	3
PASSWORD_GRACE_TIME	18

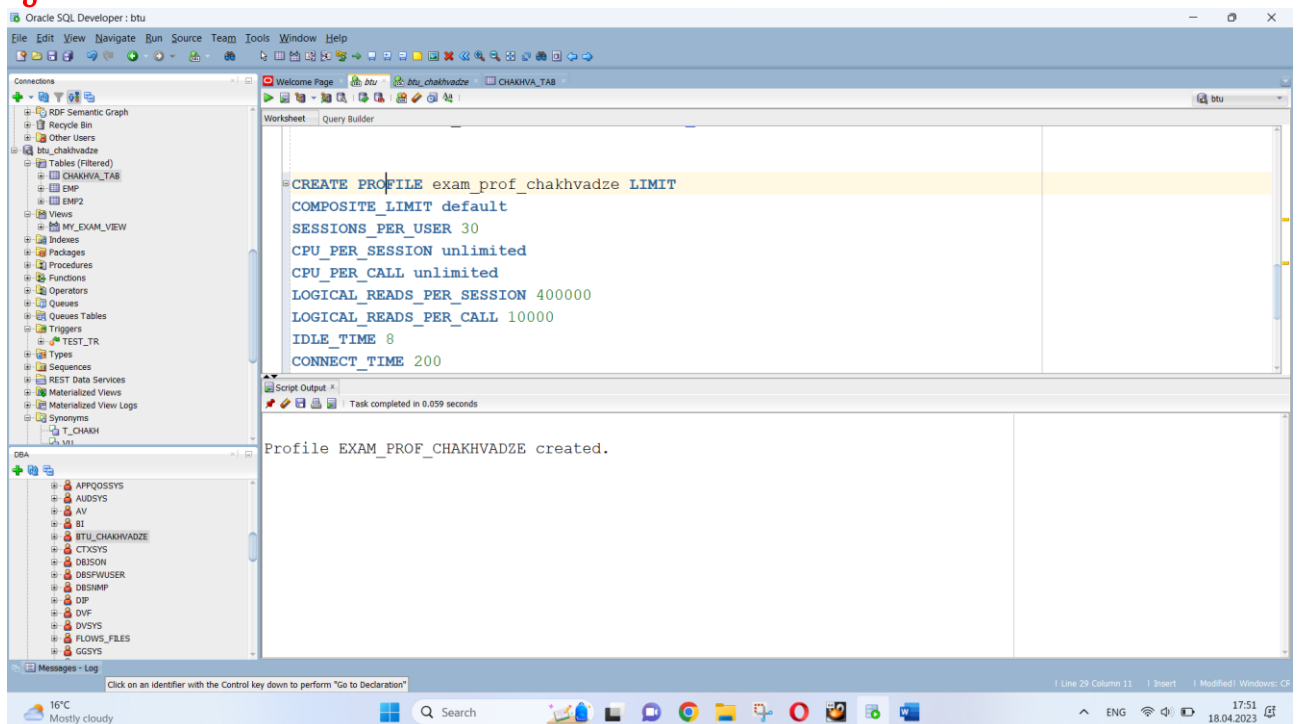
სკრიპტი:

```

CREATE PROFILE exam_prof_chakhvadze LIMIT
COMPOSITE_LIMIT default
SESSIONS_PER_USER 30
CPU_PER_SESSION unlimited
CPU_PER_CALL unlimited
LOGICAL_READS_PER_SESSION 400000
  
```

LOGICAL_READS_PER_CALL 10000
 IDLE_TIME 8
 CONNECT_TIME 200
 PRIVATE_SGA default
 FAILED_LOGIN_ATTEMPTS 20
 PASSWORD_LIFE_TIME 100
 PASSWORD_REUSE_TIME 140
 PASSWORD_REUSE_MAX 15
 PASSWORD_VERIFY_FUNCTION null
 PASSWORD_LOCK_TIME 3
 PASSWORD_GRACE_TIME 18;

სკრიპტი:

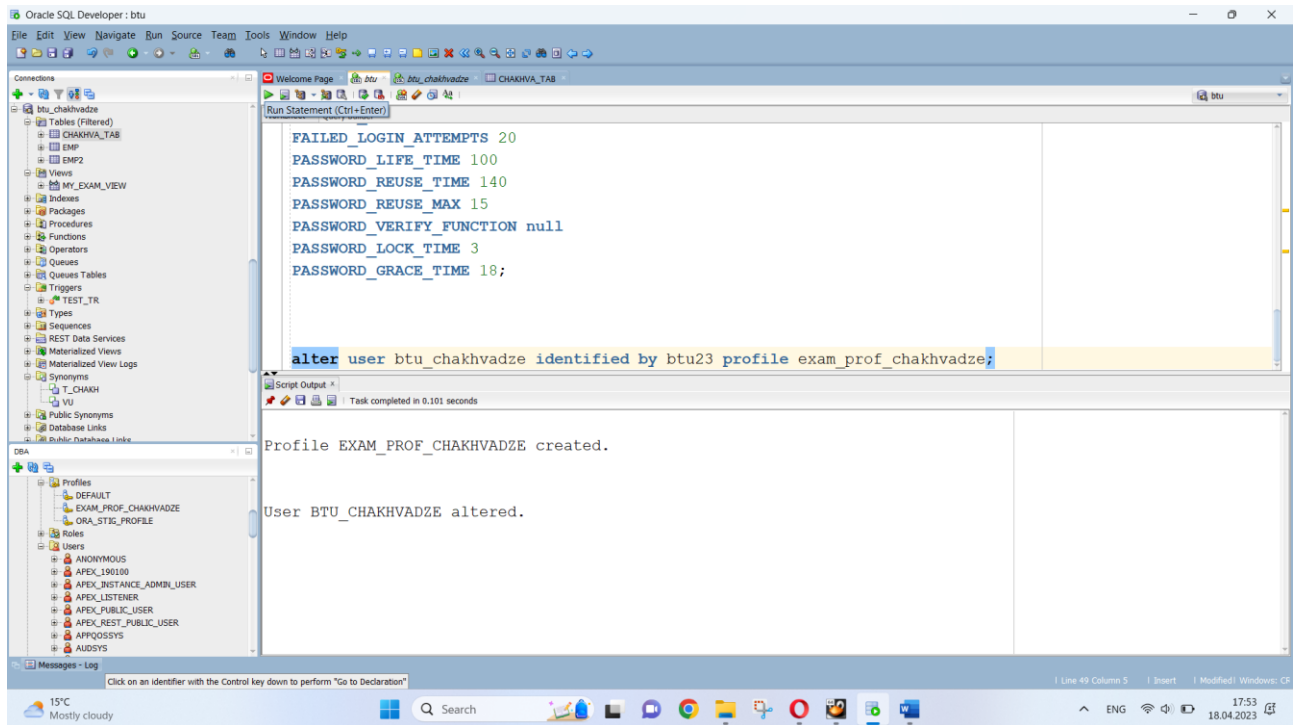


13. (1 ქულა) დამოდიფიცირეთ **btu_გვარი** იუზერის პარამეტრები პაროლი გახადეთ **btu23** და პროფილი **exam_prof_გვარი**.

სკრიპტი:

alter user btu_chakhvadze identified by btu23 profile exam_prof_chakhvadze;

სკრიპტი:



The screenshot shows the Oracle SQL Developer interface. The main window displays the following SQL statements:

```

FAILED_LOGIN_ATTEMPTS 20
PASSWORD_LIFE_TIME 100
PASSWORD_REUSE_TIME 140
PASSWORD_REUSE_MAX 15
PASSWORD_VERIFY_FUNCTION null
PASSWORD_LOCK_TIME 3
PASSWORD_GRACE_TIME 18;

alter user btu_chakhvadze identified by btu23 profile exam_prof_chakhvadze;

```

The script output window shows the following messages:

```

Task completed in 0.101 seconds

Profile EXAM_PROF_CHAKHVADZE created.

User BTU_CHAKHVADZE altered.

```

The left sidebar shows the database structure, including the user 'btu_chakhvadze' and the profile 'EXAM_PROF_CHAKHVADZE'.

14. (1 ქულა) გააქტიურე აუდიტი **btu_გვარი** იუზერის განხორციელებულ ნებისმიერ ბრძანებაზე. ნახე აუდიტის ლოგი. აუდიტი გააუქმე.

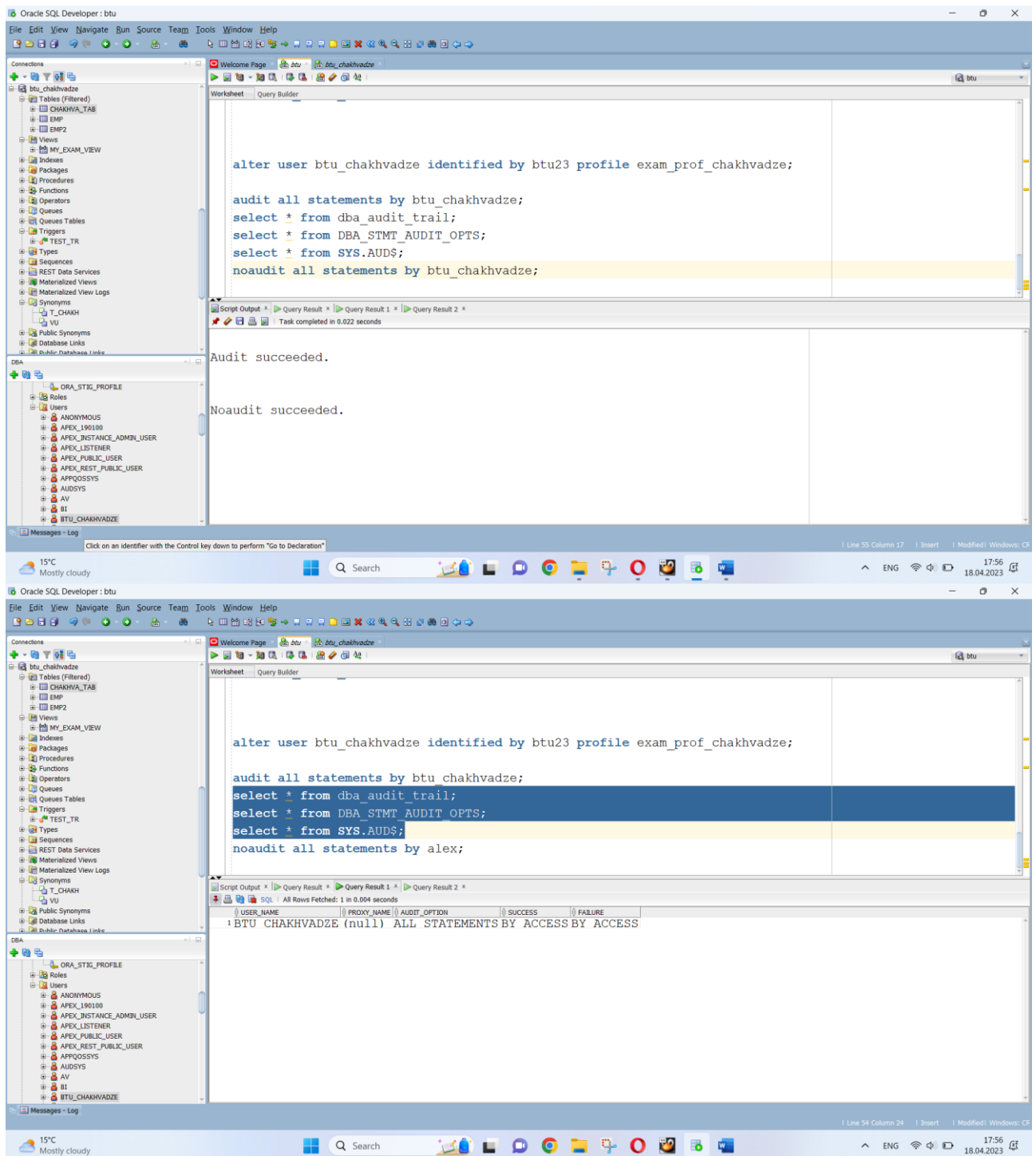
სკრიპტი:

```

audit all statements by btu_chakhvadze;
select * from dba_audit_trail;
select * from DBA_STMT_AUDIT_OPTS;
select * from SYS.AUD$;
noaudit all statements by btu_chakhvadze;

```

სკრინი:



15. (1 ქულა) წაშალეთ იუზერი **btu_გვარი**. წაშალეთ როლი **გვარი_role**. წაშალეთ პროფილი **exam_prof_გვარი**. გადაამოწმეთ არსებობს თუ არა ეს იუზერი.

სკრიპტი:

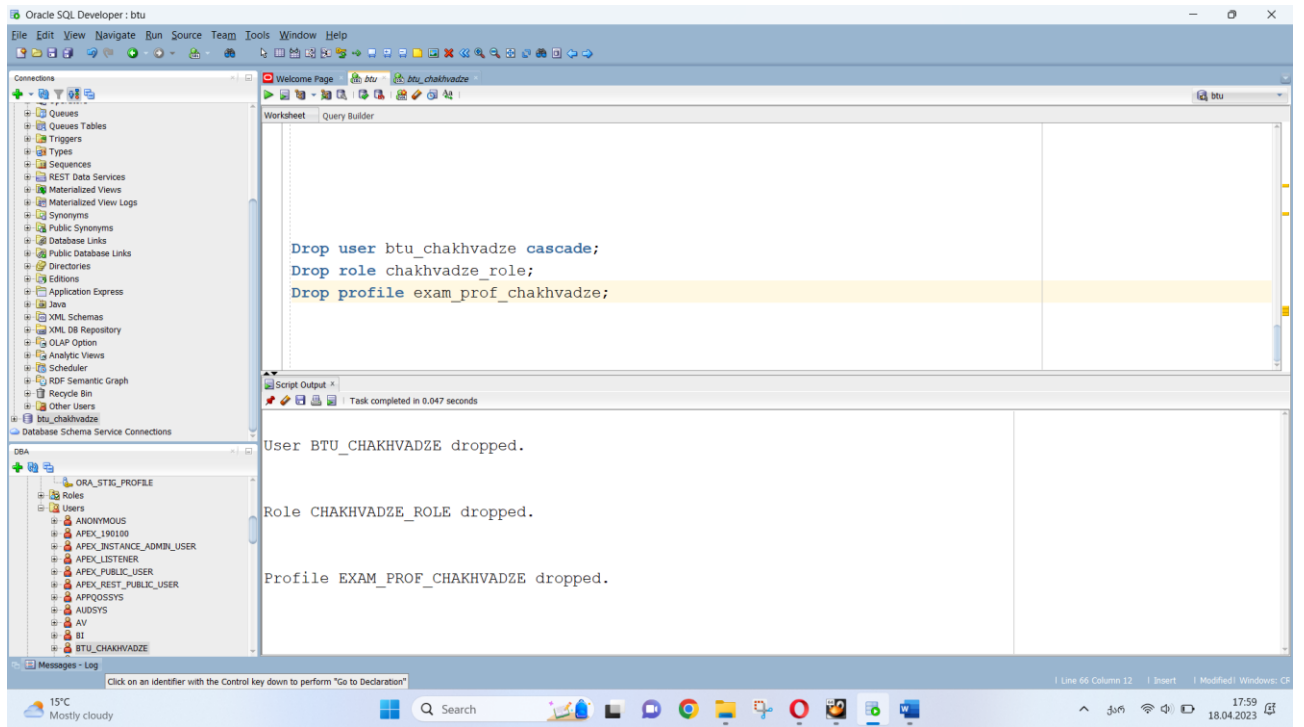
Drop user btu_chakhvadze cascade;

Drop role chakhvadze_role;

Drop profile exam_prof_chakhvadze;

select * from dba_users where username = 'BTU_CHAKHVADZE';

სკრიპტი:



The screenshot shows the Oracle SQL Developer interface. The left pane displays the 'Database Schema Service Connections' tree, with 'btu_chakhvadze' selected. The main workspace contains the following SQL script:

```
Drop user btu_chakhvadze cascade;
Drop role chakhvadze_role;
Drop profile exam_prof_chakhvadze;
```

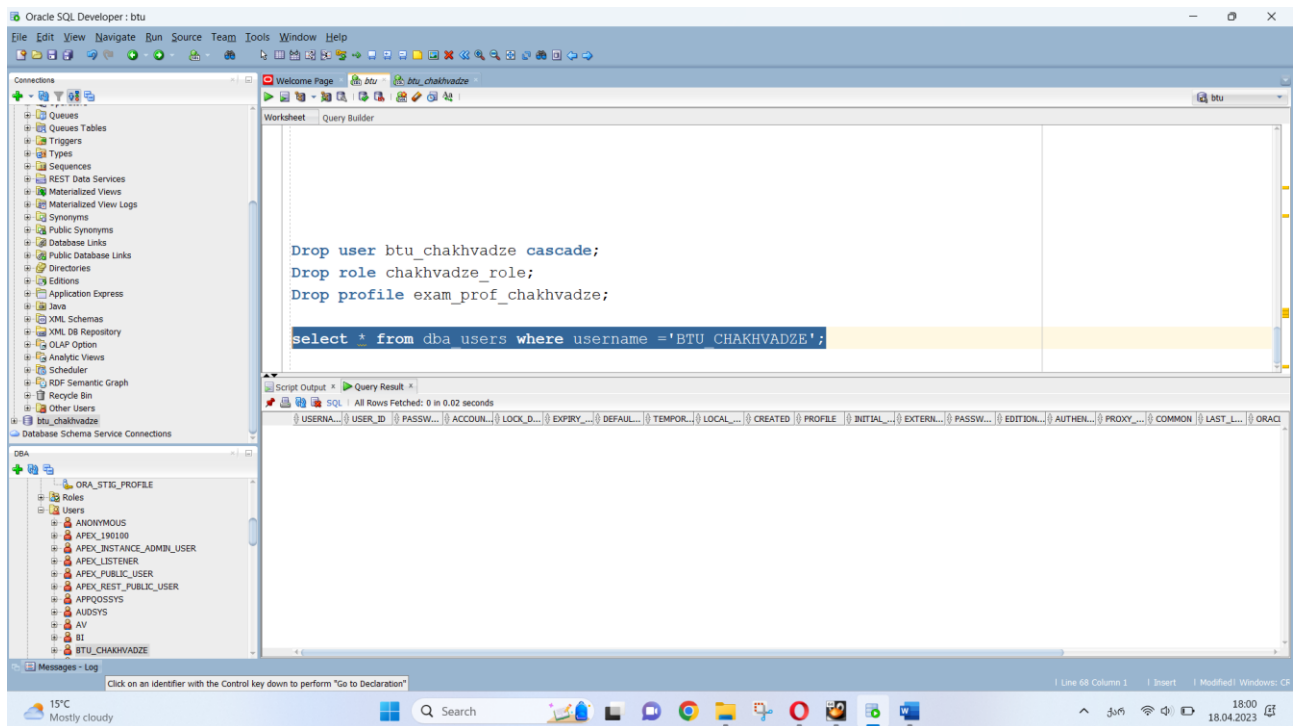
The 'Script Output' pane at the bottom shows the results of the execution:

```
User BTU_CHAKHVADZE dropped.

Role CHAKHVADZE_ROLE dropped.

Profile EXAM_PROF_CHAKHVADZE dropped.
```

The status bar at the bottom indicates 'Task completed in 0.047 seconds'.



The screenshot shows the Oracle SQL Developer interface. The left pane displays the 'Database Schema Service Connections' tree, with 'btu_chakhvadze' selected. The main workspace contains the following SQL script:

```
Drop user btu_chakhvadze cascade;
Drop role chakhvadze_role;
Drop profile exam_prof_chakhvadze;

select * from dba users where username = 'BTU_CHAKHVADZE';
```

The 'Script Output' pane at the bottom shows the results of the execution:

```
SQL All Rows Fetched: 0 in 0.02 seconds
```

The status bar at the bottom indicates 'All Rows Fetched: 0 in 0.02 seconds'.